Title:
Mesh Segmentation via Geodesic Curvature Flow

Authors:
Stephen Baek, stephen-baek@uiowa.edu, University of Iowa
Ramy Harik, harik@cec.sc.edu, University of South Carolina

Keywords:
Geodesic Curvature Flow, Geometric Flow, Surface Metric, Shape Segmentation

DOI: 10.14733/cadconfP.2017.332-336

Introduction:

Segmentation of geometry data is one of the fundamental problems in computer-aided design and geometry modeling. The problem can be briefly stated as a task of finding a partition S of a geometry X. Mathematically, a partition S of a set X is a disjoint collection of nonempty and distinct subsets of X such that each member of X is a member of some, and hence, exactly one member of S [12]. Intuitively, there can be more than one such a collection for a given X, and the problem of geometry data segmentation is, hence, to find the most *perceptually sound* partition of a given geometry X. Albeit ambiguous, the "perceptually sound" segmentation is, in general, defined based on a certain similarity metric depending on the application such that visually similar and contiguous members of X belong to the same subset of S.

In this regard, a large variety of computational methods has been proposed so far. Among those different approaches, one of the key challenges they share in common is how to define the similarity metric between the members of X. For example, Katz *et al.* [6] used the dihedral angle between two adjacent facets in order to define the likeliness of the two facets belonging to the same segment. Similarly, Page *et al.* [9] used the principal curvature values as a metric to set the threshold for the watershed clustering method. More recently, distance measures that utilizes manifold-intrinsic operators, such as the Laplace Beltrami operator, were reported to be more robust, especially for the segmentation tasks involving the isometric deformations [1],[5],[11],[14]. In this paper, we propose a novel method for deriving a shape-aware surface metric using the geodesic curvature flow (GCF). We find the GCF has a property of evolving the distance metric in a way that is more preferable for the segmentation tasks.

The Method:

A brief overview of the method is as follows. First, the geodesic distance between every pair of vertices on an input mesh X is computed as our initial surface metric. We then evolve the surface metric via the GCF in order to achieve better measurement for the clustering task. Using the eigenfunctions of the new metric, we find the spectral embedding of X. Finally, we cluster the vertices on the spectral configuration, which gives a visually intuitive segmentation of X. The new metric essentially is an indicator of how likely two distinct points are in the same segment. Hence, the spectral embedding will lead to a projection of X onto a higher dimensional space such that the distances between the points are reconfigured according to their likeliness of belonging to the same segment. Relevant literatures report that well-known metrics (e.g., Euclidean distance, geodesic distance) perform well for such a segmentation method based on a spectral embedding [7-8], but we found that a new metric achieved by evolving the geodesic distance via the geodesic curvature flow performs much better than the conventional metrics.

Geodesic Curvature Flow
The geodesic curvature flow (GCF) is a geometric flow, or informally, a continuous evolution of a curve that minimizes the arc length of a curve. Given a closed self-avoiding rectifiable curve γ lying on a differential d-manifold $\mathfrak{M}$ embedded in $\mathbb{R}^n$ ($d \le n$), the energy functional of the GCF is defined as follows:

$$E(\gamma) = \int_\gamma dl \tag{2.1}$$

Here, we restrict our curve to be rectifiable in order to make sure that it is integrable. A curve γ on a manifold $\mathfrak{M}$ is said to be *rectifiable* if and only if the length of every geodesic polygon formed by vertices $\gamma(t_1),\dots,\gamma(t_n)$, $0 \le t_1 < \cdots < t_n \le 1$ can be bounded from above by the length of the curve for some parameterization $\gamma(t)$, $t \in [0,1]$ and under the induced metric of $\mathfrak{M}$. This consequently means that the curve γ is a function with bounded variations, and thus integrable.

In a level set formulation, the energy functional in Eqn. (2.1) is converted from a line integral to a surface integral on a manifold by the coarea formula [4]:

$$E(\gamma) = \int_{\mathfrak{M}} \delta(\phi)\left|\nabla\phi\right| dA \tag{2.2}$$

where ϕ is a level set formulation of the curve γ such that the contour of $\phi = 0$ is equal to γ. δ is the Dirac's delta function. Consequently, the energy functional in Eqn. (2.2) can further be reduced to the Euler-Lagrange partial differential equation:

$$-\nabla \cdot \frac{\nabla\phi}{\left|\nabla\phi\right|}\delta(\phi) = 0\,, \tag{2.3}$$

$$\left.\frac{\partial\phi}{\partial n}\right|_{\partial\mathfrak{M}} = 0 \tag{2.4}$$

where $\partial\mathfrak{M}$ is the boundary of $\mathfrak{M}$ and n is the outward normal at the boundary. For closed $\mathfrak{M}$, the boundary condition is ignored automatically.

Further, for the discretization, we introduce a so-called "smoothed out" delta function, $\delta(\phi) = \left|\nabla\phi\right|$ as like in the standard level set methods to obtain the following gradient descent flow:

$$\frac{\partial\phi}{\partial t} = \nabla \cdot \frac{\nabla\phi}{\left|\nabla\phi\right|}\left|\nabla\phi\right|, \tag{2.5}$$

Time integration of Eqn. (2.5) provides us the "evoloved" level set function $\phi(t)$ of the original function ϕ_0. Fig. 1 shows such an evolution of a level set function defined on a human model. An interesting behavior of the GCF is that it "diffuses" and smooths out the level set function except for the narrow necks of the manifold. This property can also be observed from Fig. 1, in which the level set function is smoothed out, and hence, the function value does not change much over the large, continuous areas; whereas the level set contours are converged around relatively narrow parts such as neck, wrists, knees, ankles, waist, and so on, and hence, the function value changes relatively faster. Therefore, if the level set function was a surface distance from a certain point p (the top of the head in Fig. 1), then the evolved function under the GCF would be a better metric for the segmentation tasks, which is our insight for the proposed method in this paper.

Geometry-Aware Metric via Geodesic Curvature Flow
As aforementioned, one of the characteristics of the GCF is that it tends to evolve faster on large, continuous areas and significantly slower on narrow areas. Especially, level set curves under the GCF tend to converge near the shortest homotopic cycles. Our key insight here is to exploit such a characteristic of the GCF to evolve the distance metric on the surface in aware of the geometric contiguity.

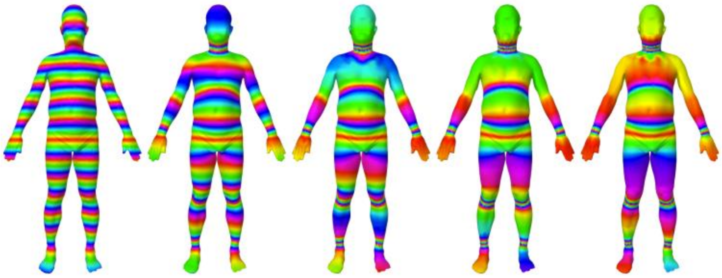

Fig. 1: Evolution of a level set function under the geodesic curvature flow (from left to right).

To achieve so, we first start with the geodesic distance $d(p,x)$ from a given point p on the manifold surface $\mathfrak{M}$. We then substitute $\phi(x)=d(p,x)$ in Eqn. (2.5) and integrate along certain time t to obtain a new distance function g:

$$g(p,x) := \phi_t(x) = \int_0^t \nabla \cdot \frac{\nabla\phi}{|\nabla\phi|}|\nabla\phi|\,dt \tag{2.6}$$

Note here that the initial computation of the geodesic distance function d does not have to be the exact geodesics, since d converges to g under the GCF in a fairly robust manner despite of small minor variations of the function values. This allows the use of fast approximate methods for the computation of the geodesic distance, such as [3], or even, the simple Euclidean distance.

For the spatial discretization, we simply assumed that the function value changes linearly on each of the triangular facets in the mesh. Based on this, gradient and divergence operators are defined as finite difference operators similar to the ones used in [3]. For the time discretization, we used the implicit Euler method.

We repeat this process for every vertex v_i in the mesh to obtain a distance matrix G whose elements are $G(i,j)=g(v_i,v_j)$. The distance matrix G is initially not symmetric, since there is no mechanism of restricting the GCF to retain the symmetry $g(x,y)=g(y,x)$. Hence, we make G symmetric simply by updating it to $G \leftarrow \frac{1}{2}(G+G^T)$ in favor of computational simplicity, where G^T is the matrix transpose of G .

Spectral Embedding and Clustering

Spectral embedding is a commonly used technique for projecting a manifold embedded in $\mathbb{R}^n$ to a different mathematical space S , with applications such as nonlinear dimensionality reduction [13], surface parameterization [15], and so on. Similarly to [8], we utilize spectral embedding to facilitate the clustering task. To do so, we first compute the affinity matrix A using the Gaussian kernel:

$$A(i,j) = e^{-G^2(i,j)/2\sigma^2} \tag{2.7}$$

Note here that $A(i,j)$ varies, by definition, in the range $(0,1]$ depending on the likeliness between two vertices v_i and v_j. In addition, from the inherent nature of the Gaussian kernel, $A(i,j)$ drops significantly towards zero when $G(i,j)>2\sigma$. Therefore, we simply cutoff values outside 2σ to zero to achieve a highly sparse affinity matrix $A(i,j)$, which has a significant numerical advantage for the computation over the original dense matrix.

The affinity matrix is then normalized to $N=D^{-1/2}AD^{-1/2}$ in order to eliminate the effect of different vertex densities, where D is a diagonal matrix each of whose elements is the sum of the corresponding row of A , i.e., $D(i,i)=\sum_j A(i,j)$.

Finally, we compute the spectral embedding of the mesh by performing the eigendecomposition of the normalized affinity matrix N. That is, when we write $\lambda_1, \lambda_2, \ldots, \lambda_K$ as the K-largest eigenvalues eigenvalues of N and $e_1, e_2, \ldots, e_K$ as their associated eigenvectors, the spectral embedding of the mesh is then represented as $Y = \Lambda^{1/2} E$, where Λ is a K-by-K diagonal matrix whose elements are $\lambda_1, \lambda_2, \ldots, \lambda_K$ and $E = \left[e_1 \quad e_2 \quad \cdots \quad e_K \right]$. Here, Y can be thought of as a new coordinate matrix in K-dimensional space for the vertices of the mesh. The new embedding, Y brings the similar points closer while it pulls the dissimilar points further apart [2]. Therefore, the clustering task in the spectral embedding is easier and more robust than that in the original embedding.

Result:

Using the proposed method, we computed segmentation for a number of benchmark models. Fig. 2 shows the result of segmentation performed on the benchmark models using the GCF metric, in comparison with the other distance metrics. …

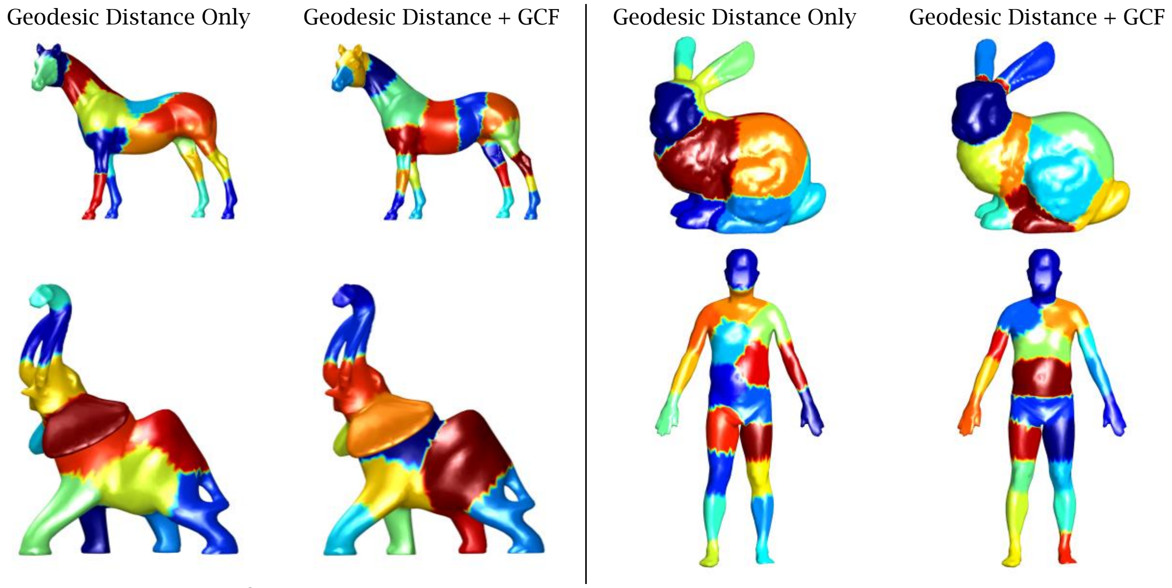

Fig. 2: Comparison of the spectral segmentation method with the geodesic distance metric (left columns) and the improved metric via the GCF (right columns).

Conclusion:

In this paper, we presented a novel method for the segmentation of a 3D mesh that utilizes the GCF. The GCF-induced metric was preferable for the segmentation tasks in a sense that it provides a better *shape-awareness* for the dissimilarity measure. In addition, even though we demonstrated the results only on the triangular meshes, the method can be generalized to any 3D geometry domain, since our formulation does not assume any particular domain. Instead, as long as there is a well-defined differential operators (i.e., gradient and divergence), our method can be seamlessly scaled to different domains including the parametric surfaces, point clouds, and polygonal meshes (see e.g., [3]).

Despite of the satisfactory performance of our method, it could struggle for the tasks that requires the surface segmentation with respect to small ridges and valleys, as the GCF tends to ignore such features. In this regard, a mathematical insight to improve our method is to introduce an additional weight function β to the GCF energy $E(\gamma)$. For the GCFs, the weight function acts as a "stopping function" that makes the flow relatively slower than the other areas. Therefore, by introducing β composed of the terms related to the principal curvature values, we could improve the

Proceedings of CAD'17, Okayama, Japan, August 10-12, 2017, 332-336

distance metric even further, which will be our future work. In addition, since the new metric possesses a good shape-awareness, a shape descriptor that encodes the geometric characteristics into a set of numerical values could be developed in the similar spirit of e.g., [10].

References:

[1] Aubry, M.; Schlickewei, U.; Cremers, D.: Pose-consistent 3D shape segmentation based on a quantum mechanical feature descriptor, In Proceedings of Joint Pattern Recognition Symposium, 2011, 122-131.

[2] Brand, M.; Huang, K.: A unifying theorem for spectral embedding and clustering, TR2002-42, Mitsubishi Electric Research Laboratories, Cambridge, MA, 2002, http://www.merl.com/publications/docs/TR2002-42.pdf.

[3] Crane, K.; Weischedel, C.; Wardetzky, M.: Geodesics in heat: a new approach to computing distance based on heat flow, ACM Transactions on Graphics, 32(5), 2013, Article No. 152. https://doi.org/10.1145/2516971.2516977

[4] Federer, H.: Geometric Measure Theory, Classics in Mathematics, Springer-Verlag Berlin Heidelberg, 1996.

[5] Harik, R.; Shi, Y.; Baek, S.: Shape Terra: mechanical feature recognition based on a persistent heat signature, Computer-Aided Design & Applications, 14(2), 2017, 206-218. https://doi.org/10.1080/16864360.2016.1223433

[6] Katz, S.; Tal, A.: Hierarchical mesh decomposition using fuzzy clustering and cuts, ACM Transactions on Graphics, 22(3), 2003, 954-961. https://doi.org/10.1145/882262.882369

[7] Liu, R.: Spectral Mesh Segmentation, Ph.D. Thesis, Simon Fraser University, Burnaby, BC, Canada, 2009.

[8] Liu, R.; Zhang, H.: Segmentation of 3D meshes through spectral clustering, In Proceedings of the 12th Pacific Conference on the Computer Graphics and Applications (PG'04), 2004, 298-305. https://doi.org/10.1109/PCCGA.2004.1348360

[9] Page, D.L.; Koschan, A.F.; Abidi, M.A.: Perception-based 3D triangle mesh segmentation using fast marching watersheds, Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR), 2003. https://doi.org/10.1109/CVPR.2003.1211448

[10] Rustamov, R.: Laplace-Beltrami eigenfuctions for deformation invariant shape representation, In Proceedings of the 5th Eurographics Symposium on Geometry Processing (SGP'07), 2007, 225-233. https://doi.org/10.2312/SGP/SGP07/225-233

[11] Skraba, P.; Ovsjanikov, M.; Chazal, F.; Guibas, L: Persistence-based segmentation of deformable shapes, In Proceedings of IEEE Conference on Computer vision and Pattern Recognition Workshops (CVPRW), 2010, 45-52. https://doi.org/10.1109/CVPRW.2010.5543285

[12] Stoll, R.R.: Set Theory and Logic, Dover Publications, Inc., New York, NY, 1979.

[13] Tenenbaum, J.B.; De Silva, V.; Langford, J.C.: A global geometric framework for nonlinear dimensionality reduction, Science, 290(5500), 2000, 2319-2323. https://doi.org/10.1126/science.290.5500.2319

[14] Yi, F.; Sun, M.; Kim, M.; Ramani, K.: Heat-mapping: a robust approach toward perceptually consistent mesh segmentation. In Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2011, 2145-2152.

[15] Zhou, K.; Snyder, J.; Guo, B.; Shum, H.-Y.: Iso-charts: stretch-driven mesh parameterization using spectral analysis, In Proceedings of the 2004 Eurographics/ACM SIGGRAPH Symposium on Geometry Processing (SGP'04), 2004, 45-54. https://doi.org/10.1145/1057432.1057439

Proceedings of CAD'17, Okayama, Japan, August 10-12, 2017, 332-336